Multimedia Authoring: Scripting (Lingo)

Cast/Score/Scripting paradigm.

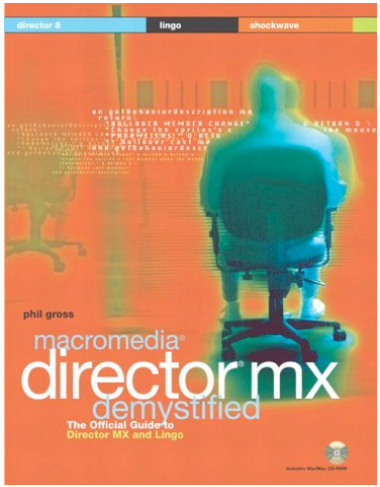
This section is a very brief introduction to Director.

For further Information, You should consult:

- *Macromedia Director Using Director* Manual — *In Library*
- *Macromedia Director : Lingo Dictionary* Manual — *In Library*
- *Macromedia Director: Application Help* — Select *Help* from within the Director application. This is very thorough resource of information.
- *Macromedia Director Guided tours* — see Help menu option.
- A variety of web sites contain director tutorials, hints and information including http://www.macromedia.com

More Director References

- *Macromedia Director MX Demystified*, Phil Gross, Macromedia Press (ISBN: 0321180976)
- *Macromedia Director MX and Lingo: Training from the Source* Phil Gross, Macromedia Press (ISBN: 0321180968)
- *Director 8 and Lingo (Inside Macromedia)*, Scott Wilson, Delmar (ISBN: 0766820084)



Related Additional Material and Coursework

Tutorials with additional Director Instructional Material

See Lab Worksheets 1 + 2

Also Assessed Exercise 2

Director Overview/Definitions

movies — Basic Director Commodity:
interactive multimedia pieces that can include

- animation,
- sound,
- text,
- digital video,
- and many other types of media.
- link to external media

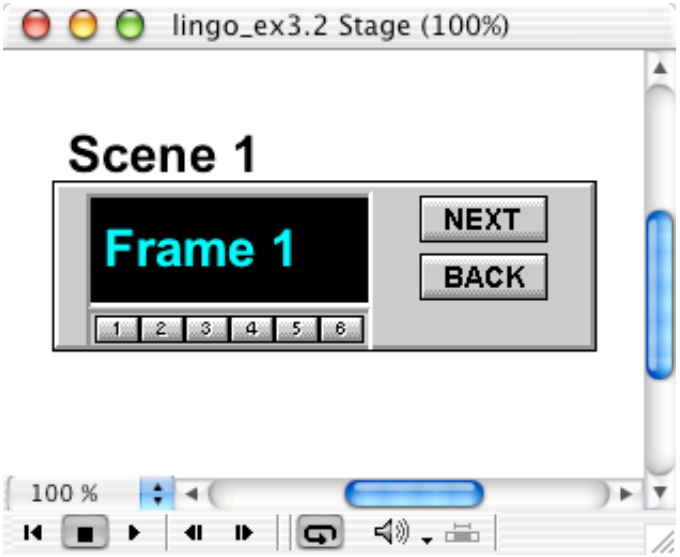
A movie can be as small and simple as an animated logo or as complex as an online chat room or game.

Frames — Director divides lengths of time into a series of *frames*, *cf.* celluloid movie.

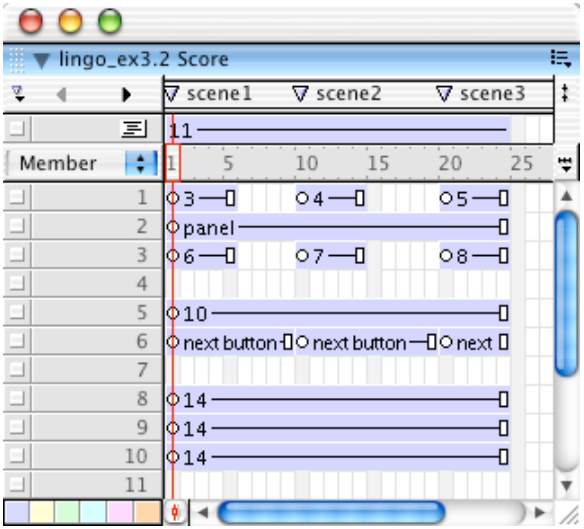
Creating and editing movies

4 Key Windows:

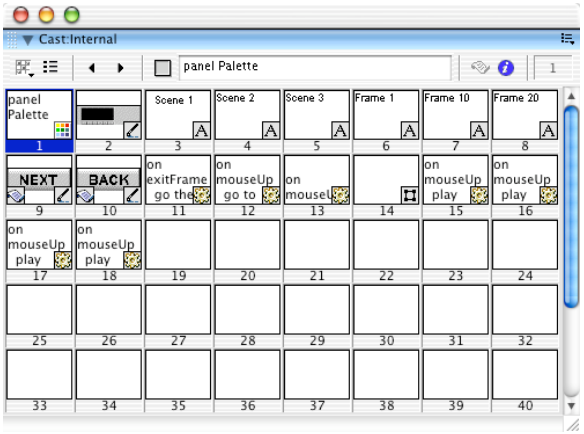
the Stage — Rectangular area where the movie plays



the Score : Where the movie is assembled;



one or more Cast windows — Where the movie's media elements are assembled;

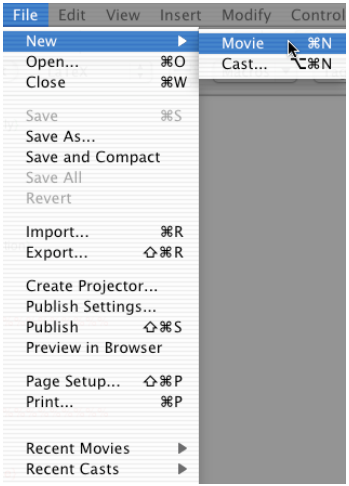


and

the Control Panel — Controls how the movie plays back.

To create a new movie:

- Choose File > New > Movie



Some other key Director Components (1)

Channels – the rows in the Score that contain sprites for controlling media

- numbered
- contain the sprites that control all the visible media
- Special effects channels at the top contain behaviors as well as controls for the tempo, palettes, transitions, and sounds.

Sprites —

Sprites are objects that control when, where, and how media appears in a movie.

Some other key Director Components (2)

Cast members —

- The media assigned to sprites.
- media that make up a movie.
- includes bitmap images, text, vector shapes, sounds, Flash movies, digital videos, and more.

Lingo — Director's scripting language, adds interactivity to a movie.

Behaviors — pre-existing sets of Lingo instructions.

Markers — identify fixed locations at a particular frame in a movie.

Lingo Scripting (1)

Commands — terms that instruct a movie to do something while the movie is playing. For example, go to sends the playback head to a specific frame, marker, or another movie.

Properties — attributes that define an object. For example `colorDepth` is a property of a bitmap cast member,

Functions — terms that return a value. For example, the date function returns the current date set in the computer. The key function returns the key that was pressed last. Parentheses occur at the end of a function,

Keywords — reserved words that have a special meaning. For example, end indicates the end of a handler,

Lingo Scripting (2)

Events — actions that scripts respond to.

Constants — elements that don't change. For example, the constants TAB, EMPTY, and RETURN always have the same meaning, and

Operators — terms that calculate a new value from one or more values. For example, the add operator (+) adds two or more values together to produce a new value.

Lingo Data Types

Lingo supports a variety of data types:

- references to sprites and cast members,
- (Boolean) values: TRUE and FALSE ,
- strings,
- constants,
- integers, and
- floating-point numbers.

Standard Program structure syntax

Lingo Script Types (1)

Director uses four types of scripts.

Behaviors — Behaviors are attached to sprites or frames in the Score.

Figure 14: Behavior Icon

Movie scripts — available to the entire movie

Figure 15: Movie script icon

Lingo Script Types (2)

Parent scripts — special scripts that contain Lingo used to create child objects.

Figure 16: Parent script icon

Scripts attached to cast members — independent of the Score. don't appear in the Cast window.

Figure 17: Script button

Director Example 1: Simple Animation

A Bouncing Ball Graphic

Run Example in Browser (Shockwave)
Run Example in Browser (Lecture ONLY)

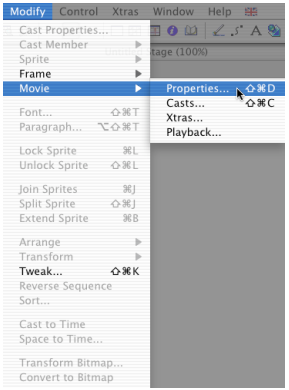
- No Lingo scripting.
- basic animation where a cast member

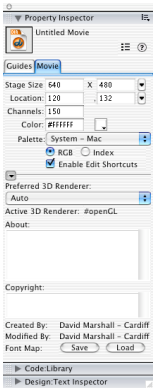
Creating the Bouncing Ball Graphic

The following steps achieve a simple bouncing ball animation along a path:

1. Let us begin by creating a new movie and setting the Stage size:

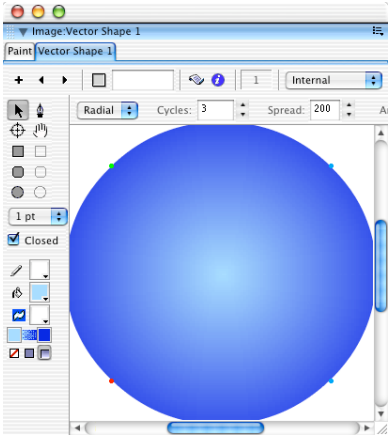
   - Start a New movie: File > New > Movie (Shortcut = Command+N)
   - Choose Modify > Movie > Properties.
     In stage size, choose 640 x 480.





2. Now let us create a ball, using a the vector shape tool:

- Choose Window > Vector Shape (Shortcut = Command+Shift+V)
- Click the filled ellipse button.
- Draw an ellipse (circle) about the size of the Vector Shape Window
- Click on the Gradient fill button.
- To change the colours, click the colour box on the left side of the Gradient colour control
- Change the colour on the right side of the Gradient Colours to a dark blue.
- Change the Gradient type pull-down menu from Linear to Radial.
- Change the Stroke Colour to white .

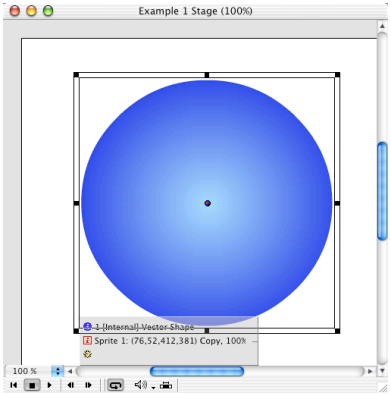


3. Now let us change a few other properties of this ellipse

- Close the Vector Shape window.
- In the Cast Window, select the ellipse.
- Choose Edit > Duplicate (Shortcut = Command+D).
- Double click the new cast, which opens it in the Vector Shape Tool.
- Change the Cycles to 3 and the Spread to 200.
- Name the latest ellipse to ’bouncing ball’

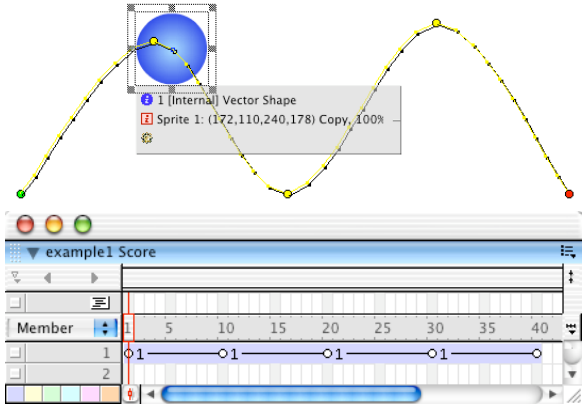
4. Now we are going to animate the ball.

- Drag 'bouncing ball' from the cast member window to the stage.
- You will notice the sprite (the object that appears in the score) is extended over 20 frames.
- Drag the right end of the sprite to frame 40.
- Click anywhere in the middle of the sprite to select it.
- resize the ellipse.



4. Ball Animation (Key Frames)

- Click on frame 40 in channel 1 (the end of the sprite), hold down Option and shift and drag the ellipse to the right end of the stage.
- To curve the path, we are going to insert keyframes within the sprite.
- Click on frame 10 of the sprite and choose Insert > Keyframe (Shortcut = Command+Option+K)
- Create keyframes at frame 20 and 30.
- at each keyframe, a circle appears on the path shown on the stage.
- Click on the keyframe 10 circle and drag it up.
- Change other Keyframes.
- Rewind and play the movie.

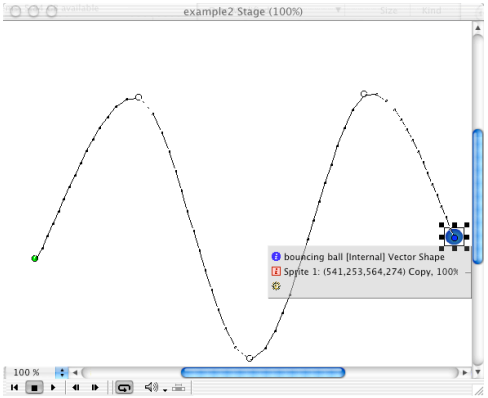


Further Animation: 1.1 Shrinking the ball

Run Example Shrinking the ball (Shockwave)

Run Shrinking the ball (Lecture Only)

- (Optional) Click on keyframe 40 in the score and drag it to frame 60, notice how all the keyframes spread out proportionally.
- (Optional) Click on the keyframes in the score and adjust the path if you feel like it.
- While moving the keyframes, resize the balls so they slowly get smaller. Notice while you resize the balls, the path changes and you will need to edit the path again.
- Rewind and play the movie.
- Save your movie as *example2.dir*.

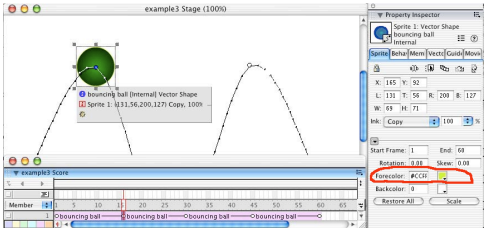


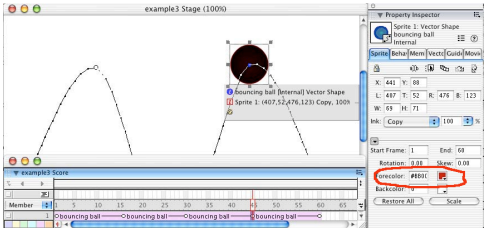
1.2. Animating sprite colour

Run Example: Animating sprite colour (Shockwave)

Run Example: Animating sprite colour (Lecture Only)

- Working still with *example1.dir*.
- Open Property Inspector for Sprite
  - Right Mouse (or Ctrl) on Sprite (Score or Stage)
  - Select **Properties...**
- Click on the keyframes in the score, and change the Foreground colour chip, *Forecolor*, to different colours.
- Changing the foreground colour is like putting a coloured film over your object. The resulting colour is a mixture of the object's original colour and the 'film'. For this reason, light colours work better than dark colours for this effect..
- Rewind and play the movie.
- Save as *example3.dir*





1.3. Animating sprite transparency — Making the Ball Disappear

Run Example: Making the Ball Disappear (Shockwave)
Run Example: Making the Ball Disappear (Lecture Only)

- Open *example1.dir*
- Open Property Inspector for Sprite
- Click on the keyframes in the score, and
- Change the Blend Transparency to 100, 75, 50, 25, 0 for the consecutive keyframes.
- Rewind and play the movie.
- Save as *example4.dir*

1.4. Animating sprite shape — Deforming The Ball

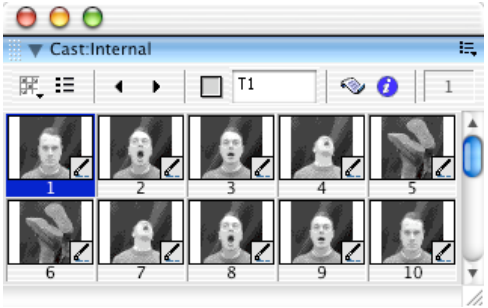
Run Example: Deforming The Ball (Shockwave)
Run Example: Deforming The Ball (Lecture Only)

- Open *example1.dir*
- Open Property Inspector for Sprite
- Click on the keyframes in the score, and
- Change the Skew Angle to 0, 20, 40, 60 and 80 for the consecutive keyframes.
- Rewind and play the movie
- Save as *example5.dir*

Director Example 2: Importing media

To import multimedia data there are two basic ways:

- Choose File > Import ...
  Useful for importing batches of data (*e.g.* Several image sequences.
- Drag and drop source media into a cast member location
  Quite Intuitive



Examples: Simple Image import and Manipulation

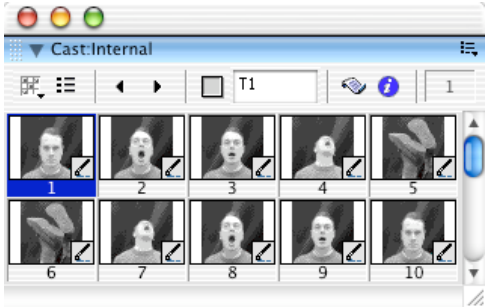
- Drag an image into a spare cast member.
- Drag this cast member to the Score
- Set suitable Properties for Sprite
  - Manipulate as for a vector item above.
- Examples:
  - ex_dave_roll.dir sets up some keyframes and alters the rotation of the image (Shockwave)
  - ex_dave_roll.dir sets up some keyframes and alters the rotation of the image (Lecture Only)
  - ex_dave_sq.dir alters the skew angle (Shockwave)
  - ex_dave_sq.dir alters the skew angle (Lecture Only)

Example: Falling Over Movie, ex_dave_movie.dir

Example: Falling Over Movie, ex_dave_movie.dir (Shockwave)

Run Example: Falling Over Movie, ex_dave_movie.dir (Lecture Only)

- Several Gif images depicting sequence exist on disk.
- Choose File > Import
- Select items you wish to import by double-clicking or pressing the Add button
- Click on the Import Button
- Several new cast members should be added
- Set looping on and play

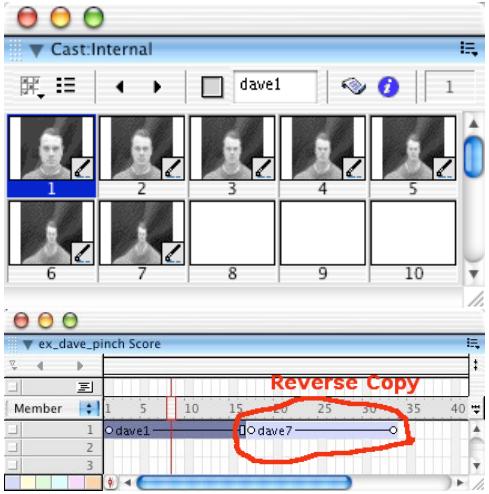


Example: Pinching Movie Movie, ex_dave_pinch.dir

Example: Pinching Movie Movie, ex_dave_pinch.dir (Shockwave)

Example: Pinching Movie Movie, ex_dave_pinch.dir (Lecture Only)

- Photoshop has been used to set a pinch effect of varying degree for an image.
- Import images as before
- To reverse the image set to obtain a smooth back and forth animation:
  - Select the sprite sequence in the score
  - Copy Sequence — Press Command+C (Copy),
  - Click on the frame just after the sprite sequence
  - Paste Sequence — press Command+V (Paste).
  - Click on this second sprite sequence and choose Modify > Reverse Sequence.
  - Select the 2 sprites by pressing Shift and clicking on both. Choose Modify > Join Sprites.



Simple Lingo Scripting

Director Example 3: Very Simple Action

Here we illustrate the basic mechanism of scripting in Director by developing and extending a very basic example:

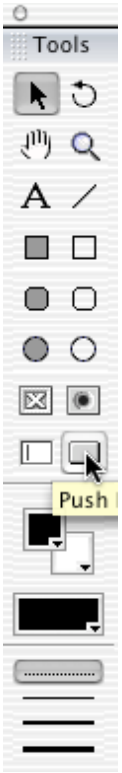
Making a button beep and attaching a message to a button

Making the a button beep (Shockwave)
Making the a button beep (Lecture Only)

Making the Button Beep Movie

- Open a new movie.
- Turn the looping on in the control panel.

- Open the tool palette.
- Click the push button icon.
- Draw a button on the stage, and type in a label: "`button`" here

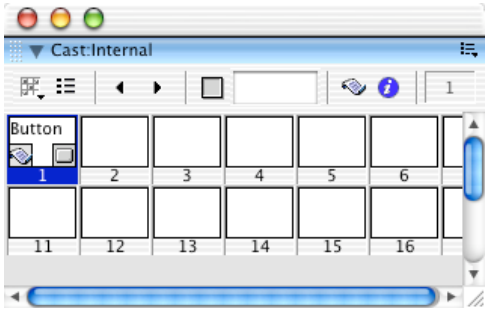


Our First Lingo

Now lets write a simple script for the button:

- Press Ctrl+click the button in the cast window and choose Cast Member Script.
- Director writes the first and last line for us, add a beep command so the script look like this:

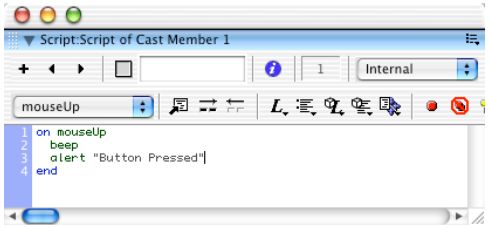
```
on mouseUp
    beep
end
```



- Close the window.
- Rewind and play the movie.
- Click the button a few times.

To pop up a message box on button press (and still beep)

- Reopen the cast member script.
- Change the text so it now reads.




```
on mouseUp
   beep
   alert "Button Pressed"
end
```

- Close the window.
- Play the movie and click the button.

Director Example 4: Controlling Navigation with Lingo

A slightly more complex Lingo Example

This examples illustrates how we may use Lingo Scripts as:

- Cast Member Scripts
- Sprite Scripts
- Behaviour Scripts

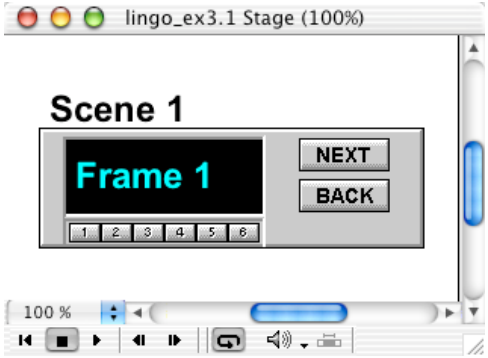
Director Example 4: Ready Made Example

To save time, we begin we a preassembled Director movie:

Run Lingo Navigation Example (Shockwave)

Run Lingo Navigation Example (Lecture Only)

- Open lingo_ex.3.2.dir
- Play the movie — press some of the buttons
  - The Numbered buttons record moves through Scenes/Frames
  - The `Next/Back` buttons replay what has been recorded.



The *Loop the Frame* script

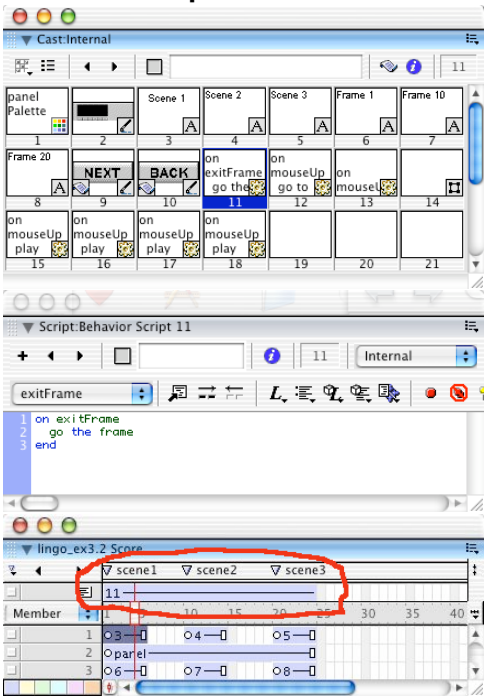
We are first going to create a *loop the frame* script:

- **Cast Member 11** controls the Frame Looping
- Note we have created a special frame **marking channel** in the Score.
- To create the associated script either
  - Double click on the script icon in the Score
  - Ctrl-click on the Cast member and select **Member Script**
- The scripting window appears. You can edit the script text, it now reads:

```
on exitFrame
   go the frame
end
```

This frame script tells Director to keep playing the same frame.

- The Loop lasts to **frame 24**



- Pressing down Alt and dragging the frame script in the Score can change this length.

Scene Markers (1)

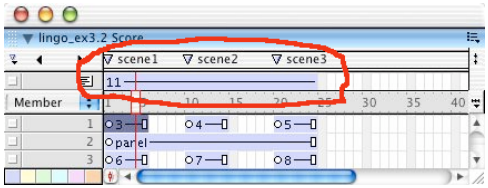
Now we will create some *markers*

- **To Create a Marker You Click in the marking channel for the Frame and label some the marker with some typed text**

In this example:

- Markers are at `frame 1`, `10` and `20`, naming them `scene1`, `scene2` and `scene3` respectively.
- **Note:** You can delete a marker by clicking the triangle and dragging it below the marker channel.
- A cast member (9) script for the **next** button has also been created:

```
on mouseUp
    go to next
end
```



- The `go to next` command tells Director to go to the next consecutive marker in the score.

Scene Markers (2)

- A cast member (10) script for the back button has also been created:

```
on mouseUp
    go to previous
end
```

  The `go to previous` command tells Director to go to the previous marker in the score.

- Once again, Play the movie, click on these buttons to see how they work.

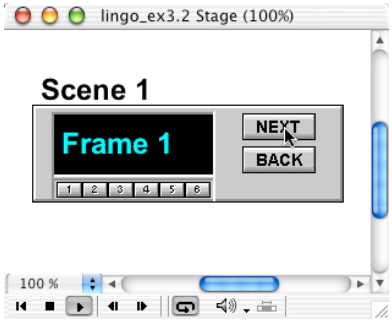
Sprite Scripts

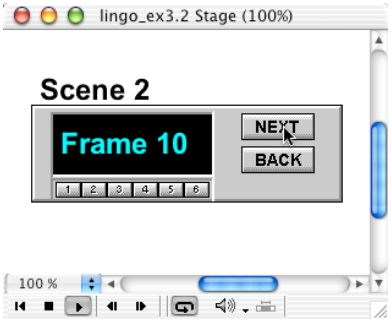
Now We will create some sprite scripts:

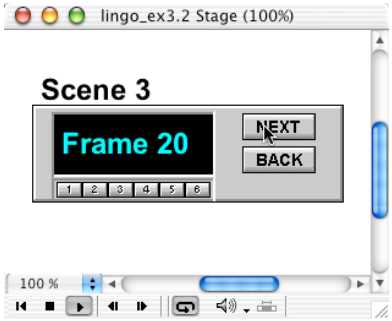
- Sometimes a button will
  - behave one way in one part of the movie and
  - behave another way in a different part of the movie.
  - A typical example use of sprite scripts.

The Next Button Sprite Scripts (1)

Desired Action of Next Button: Jump to next scene

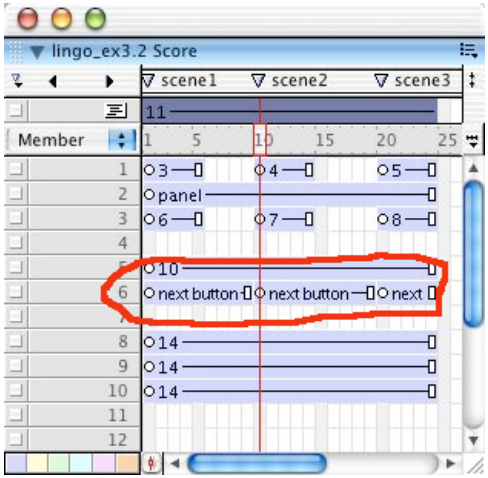


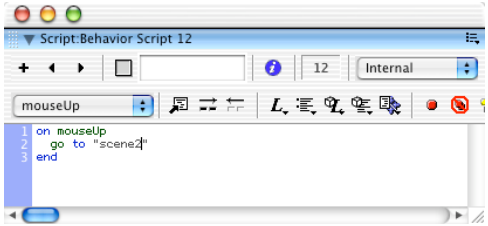




The Next Button Sprite Scripts (2)

- Here we have split actions to map to our **Scene Markers**. To achieve this:
  - Click on frame 10 of channel 6 (the next button) this sprite and choose Modify > Split Sprite.
  - Do the same at frame 20.
- To attach a script to each split action:
  - Select the each sprite sequence (here in channel 6).
  - Ctrl-click on sequence and select `Script...` from the pull-down in the score to give a script window.
  - We add a suitable jump to *next scene*
  - In example shown we have `go to "scene2"` :
    This command tells Director to send the movie to marker `"scene2"`.
  - Could do other sequences similar – But alternatives exist.





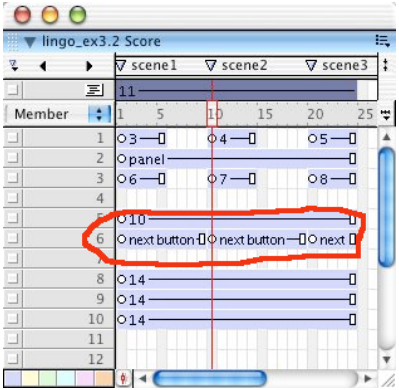
Behaviour Scripts

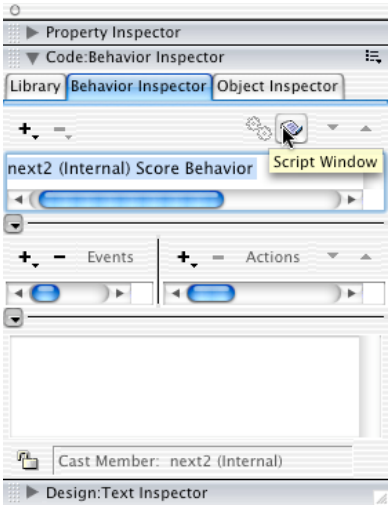
Example Here: Another way to write the sprite script on last slide - using the Behaviour Inspector.

Behaviour Scripts can do **A LOT MORE**.

A Behaviour Script for Next Button (Scene 2) (1)

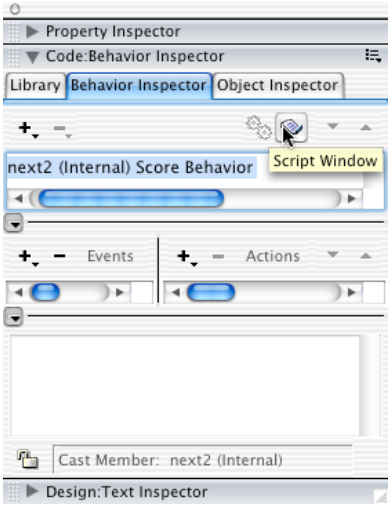
- We now work with the second sprite sequence (channel 6 in the score).
- We will create (or have created) an associated behaviour script:
  - Ctrl-click on the second sequence
  - Open the Behaviour Inspector window
  - Click on the Script Window icon next to the Behaviour Inspector Tab.
- To Create/name a new Behaviour:
  - Click the + icon at the top left of the window and select new behaviour from the pull-down.
  - Give the behaviour a name, here it is called `next2`.

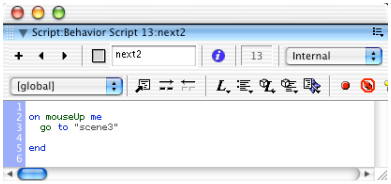




A Behaviour Script for Next Button (Scene 2) (2)

- To add events/actions to the script you can:
  - Under Events click the + icon
    In this example we have added a `mouseUp` from the menu.
  - Under Actions click the + icon
    In this example we have chosen Navigation > Go to marker then find scene3 on the list
  - You can add/edit Lingo text manually in the Script Editor window for the particular behaviour





Summary: Sprite Script Order

We now 2 scripts attached to a single object (achieving much the same task):

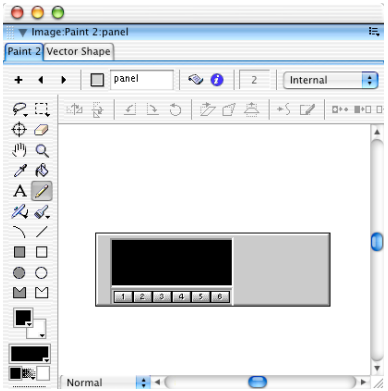
- a Cast Member script
- a Sprite script.
- Sprite scripts take priority over cast member scripts
- So Here cast member script will be ignored.

Some more Lingo to add to our example

Another part of our Application:
The jump buttons 1-3 (Buttons 4-6 currently inactive).

We will be using Lingo `play`/ `play do` to record actions

We have created a Vector Graphic Image (Cast Member 2) for the main Recorder Interface



A problem in Director?

In Director: a script can only be associated with a complete Object

For the way we have created the Recorder Interface we require (and this is clearly a common requirement in many other cases):

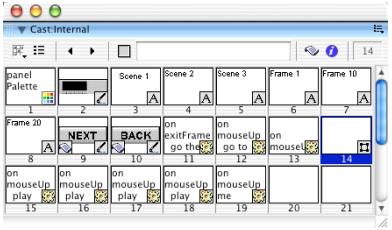
- Only part an image to be linked instead of the whole object.
- One part for each of the jump buttons 1-3.

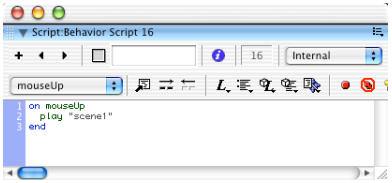
There is a solution:

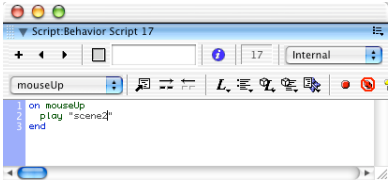
- Use **invisible buttons**.
- These are shape cast members with an invisible border.

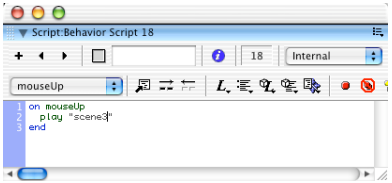
Creating our Invisible Buttons

- We have added our invisible as Cast Member 14. To create this component:
  - Open the Tool palette window.
  - Click on the no line button.
  - Click on the rectangle button and draw a rectangle on the stage around the 1 button.
- We have added this sprite to 8 and we have attached a sprite script:
  - Ctrl-click on frame 1 of channel 8 and select script.
  - Attach a sprite script to this shape with the command play ”scene1”.
  - Extend the sprite sequence so it covers frame 1 to 24.
- Repeat the steps placing the sprite over the 2 and 3 button









Final Part: the Back Button (1)

Director provides the ability to record actions for future use

The Lingo Play command

The `play` command is similar to the `go to` command but allows:

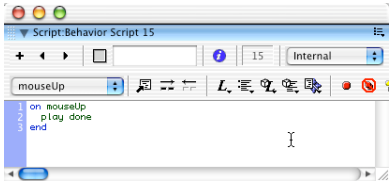
- Director *records* every time a play is initiated,
- Keeping track of the users' path through the movie.
- You can move back on along this path by using the `play done` command.

Final Part: the Back Button (1)

So in this Example

- Select the sprite sequence in channel 5 and Cast member 10.
- Attach a Sprite script reading

```
on mouseUp
    play done
end
```



- Rewind, play the movie, click all the 1, 2, 3 buttons in various orders, click the back button also and observe the effect of Back button
- Complete example: *lingo_ex3.2.dir* (Web Based)
- Complete example: *lingo_ex3.2.dir* (Local Version)